

LISTING OF CLAIMS

1-21. (Canceled)

22. (Currently Amended) A method for developing a device management application to manage a remote device, comprising:

developing providing a graphical programming language environment to develop
source code for a command-set aware user interface, the source code to be used to generate a user interface to execute at the remote device to configure a configuration parameter of the remote device according to a configuration command of the command set;

developing providing a graphical programming language environment to develop
source code for a command-set unaware graphical user interface (GUI) of the device management application ~~with a graphical programming language~~, the GUI to be executed remotely from the remote device;

linking the source code for the command-set aware user interface in the source code for the command-set unaware GUI to provide a graphical component in the GUI associated with the configuration command; and

building the source code for the command-set unaware GUI with the linked source code for the command-set aware user interface to result in the device management application having the GUI with the command-set aware user interface running beneath the GUI, remotely from the remote device, the GUI having the graphical component to provide access to the associated configuration command.

23. (Previously Presented) A method according to claim 22, wherein the command-set aware user interface comprises a user interface having code to execute every configuration command necessary to generate every configuration state of the remote device.

- 24.** (Previously Presented) A method according to claim 22, wherein the configuration parameter comprises one or more of a variable, a data structure, or a function defining a configuration state of the remote device.
- 25.** (Previously Presented) A method according to claim 22, wherein linking the source codes comprises providing a code hook in the source code for the GUI to the command-set aware user interface.
- 26.** (Previously Presented) A method according to claim 22, wherein linking the source codes comprises providing the source code for the command-set aware user interface as a library referenced in the source code of the GUI.
- 27.** (Previously Presented) A method according to claim 22, wherein linking the source codes comprises providing a macro of the programming language in the source code of the GUI.
- 28.** (Previously Presented) A method according to claim 27, wherein providing the macro comprises referencing an object-oriented class object in the source code of the GUI, the class object to invoke a routine of the source code of the command-set aware user interface.
- 29.** (Previously Presented) A method according to claim 22, wherein the source code for the command-set aware user interface comprises firmware defining the user interface, and wherein linking the source codes comprises re-using the firmware by linking the firmware to develop source code defining the device management application.
- 30.** (Previously Presented) A method according to claim 22, wherein building the source code comprises compiling the source code for the command-set unaware GUI with the linked source code for the command-set-aware user interface to generate an executable binary device management application.

31. (Previously Presented) A method according to claim 22, further comprises:

identifying an updated configuration command in the configuration command set at the remote device;

updating the source code for the command-set aware user interface to reflect the updated configuration command in the configuration command set; and

rebuilding the source code for the command-set unaware GUI with the linked updated source code for the command-set aware user interface to result in an updated device management application having a graphical component to provide access to the updated configuration command.

32. (Previously Presented) An article of manufacture comprising a computer-readable medium having instructions to cause a computer to perform operations including:

receiving source code defining a console user interface (CUI) to generate a CUI to execute at a network device, the CUI to configure a configuration kernel (CK) of the network device according to a configuration command;

receiving source code defining a graphical user interface (GUI) to generate a device management application, the GUI to be executed at a management point remote from the network device, the source code for the GUI to include a hook to the source code for the CUI to provide a graphical component in the GUI to operate a function of the CUI to access the configuration command from GUI; and

building the source code for the GUI with the hook to the source code for the CUI to create the device management application having the CUI running under the GUI, remotely from the remote device, the GUI having the graphical component to provide access to the associated configuration command.

- 33.** (Previously Presented) An article of manufacture according to claim 32, wherein the source code for the GUI to include a hook to the source code for the CUI comprises the source code for the GUI to include a macro referencing the source code for the CUI.
- 34.** (Previously Presented) An article of manufacture according to claim 32, wherein the operations further include providing the source code defining the CUI as a library to the source code defining the GUI, and wherein the source code for the GUI to include a hook to the source code for the CUI comprises the source code for the GUI to include a reference to a class defined in the library.
- 35.** (Previously Presented) An article of manufacture according to claim 32, wherein the source code defining the CUI comprises firmware code defining the CUI, and wherein receiving the source code defining the CUI comprises re-using the firmware code to link the firmware code with the source code defining the GUI to develop the device management application.
- 36.** (Previously Presented) An article of manufacture according to claim 32, wherein the operations further include:
- identifying an updated configuration command in the configuration command set at the network device;
 - receiving updated source code defining the GUI to reflect the updated configuration command in the configuration command set; and
 - rebuilding the source code for the GUI with the hook to the updated source code for the command-set aware user interface to result in an updated device management application having a graphical component to provide access to the updated configuration command.

- 37.** (Previously Presented) An apparatus for managing a remote device, comprising:
a memory having a management application, including:
a graphical user interface (GUI) module having a graphical component associated with a configuration command, the graphical component responsive to a user input to operate the configuration command; and
one or more code libraries including code defining a configuration kernel (CK) and code defining a console user interface (CUI) of a remote device, the CK having a configuration parameter associated with the configuration command, the configuration parameter corresponding to a resource of the remote device, the CUI to interface the configuration kernel with the GUI, the code libraries linked with the GUI and compiled with the GUI to create the management application having the GUI with the CK and the CUI running under the GUI, remotely from the remote device;
a communications interface coupled with the remote device to communicate a configuration update for the remote device from the management application; and
a processor coupled with the memory to operate the management application, and coupled with the communications interface to provide the configuration command to the communications interface.
- 38.** (Previously Presented) An apparatus according to claim 37, wherein the GUI module further comprises a reference to a macro defined in one or more of the libraries.
- 39.** (Previously Presented) An apparatus according to claim 37, wherein the GUI module further comprises a hook to a subroutine defined in one or more of the libraries.
- 40.** (Previously Presented) An apparatus according to claim 37, wherein the code libraries including code defining the CK and defining the CUI comprises firmware code defining the CK

and firmware code defining the CUI, the firmware code developed for execution on the remote device.

41. (Previously Presented) An apparatus according to claim 37, wherein the communications interface coupled with the remote device to further receive an updated configuration command from the network device, and the processor further to update the code libraries with the updated configuration command and to re-compile the updated code libraries with the GUI to create an updated management application.